i created alert and it is reflecting to interlink and serviceNow by using postman. when i hit the same alert . then in interlink it has created at once and counting the run times but in the servicenow the same alert is duplicating but i want this as same as like interlink how can i achieve this by using alert manager.

https://support.servicenow.com/kb?id=kb\_article\_view&sysparm\_article=KB0753955&form=MG0AV3

**ChatGPT said:**

ChatGPT

To address the issue of duplicate alerts in ServiceNow when you send alerts using Postman, we need to ensure that the alert is being handled properly by ServiceNow to avoid creating duplicate records.

Here’s a structured approach to achieve the behavior you're looking for:

**1. Check for Duplicate Alerts in ServiceNow**

* ServiceNow typically uses some unique identifier (like alert\_id, incident\_id, or a custom field) to differentiate between new alerts and existing ones.
* Ensure that you are passing a unique identifier for the alert, such as a correlation ID or alert ID, so that ServiceNow can detect that the alert is the same and not create a new incident or alert each time.
* You can make use of the **Alert Manager** in ServiceNow to configure rules that prevent the creation of duplicates.

**2. Configure Alert Management to Prevent Duplicates**

* In ServiceNow, **Alert Management** can be used to prevent duplicate alerts based on predefined conditions.
* You can configure **Alert Rules** in ServiceNow to check incoming alerts and identify whether a similar alert has already been created. If a similar alert exists, you can update the existing alert instead of creating a new one.

To configure Alert Management to prevent duplicates, follow these steps:

**a. Define a Unique Identifier**

* When sending the alert through Postman, make sure to pass a unique identifier (like alert\_id, correlation\_id, or a custom field). This will help ServiceNow identify the alert as a duplicate.

**b. Create or Modify Alert Rules in ServiceNow**

* Go to **Alert Management** > **Alert Rules** in ServiceNow.
* Create a new rule or modify an existing one.
* Set conditions to check if an alert with the same unique identifier (e.g., alert\_id) already exists. If so, the system can be configured to either update the existing alert or suppress the creation of a new one.

Here’s a rough example of what the condition might look like:

* + **Condition:** If the alert\_id or correlation\_id already exists, **do not create** a new alert, but instead, **update** the existing one.

**c. Set Alert Action**

* For the action in case of a duplicate, you can choose to **update** the existing alert or **ignore** the incoming alert.

**3. Use Webhooks (Optional)**

If you're using a webhook to send the alert to ServiceNow, you can add additional logic in the webhook configuration to check for duplicate entries before sending a new request.

**4. Test the Setup**

* Once the changes are made in ServiceNow, test by sending the same alert multiple times using Postman.
* In **Interlink**, confirm that it is working as expected (i.e., the alert is logged once).
* In **ServiceNow**, verify that the same alert does not create a new entry but instead updates the existing one or is suppressed if it's a duplicate.

**Example of Preventing Duplicates in ServiceNow Alert Rules:**

* **Condition:** If alert\_id = incoming\_alert\_id AND state = active, **update the existing alert** or **ignore**.
* **Action:** If a match is found, update the existing record or suppress the alert from being created.

**5. Investigate Postman Configuration**

* Ensure that you're sending the correct payload with the right parameters when calling the ServiceNow API in Postman. Check that the unique identifier (like alert\_id) is included consistently with each alert request.

**6. Use ServiceNow’s Out-of-the-box Deduplication Features**

* ServiceNow comes with out-of-the-box features for deduplication. This might involve setting up correlation rules or adjusting the **alert rules** within ServiceNow to correlate similar alerts based on some condition (e.g., same alert type, same severity, same correlation ID, etc.).

**Conclusion:**

To prevent the duplicate creation of alerts in ServiceNow, configure the **Alert Rules** in ServiceNow to check for existing alerts using a unique identifier, and ensure that your Postman requests include this unique identifier. This way, ServiceNow will either update the existing alert or suppress it instead of creating a duplicate entry.

STEP 2:

here i want to create alert rules for reduce the duplicates

Great! To create alert rules in ServiceNow that reduce duplicates, follow these steps:

1. **Identify Unique Keys**:
   * **Ensure Unique Identifiers**: Identify fields that can serve as unique keys (like source, event type, and timestamp) to differentiate alerts. These keys help ServiceNow identify whether an incoming alert is a duplicate.
2. **Navigate to Alert Management**:
   * Go to the **Alert Management** module in ServiceNow.
3. **Create a New Alert Rule**:
   * Click on **Create New** to define a new alert rule.
4. **Define Rule Criteria**:
   * **Set Conditions**: Specify the conditions that will trigger an alert. For instance, you might define an alert to trigger only if a certain error message occurs, or if a specific metric exceeds a threshold.
   * **Use Unique Keys**: Make sure to include your unique keys in the rule criteria.
5. **Set Correlation Rules**:
   * **Group Similar Alerts**: Use correlation rules to group similar alerts. This can be done by defining rules that check for similar source events within a certain time frame.
   * **Suppress Duplicates**: Configure the rule to suppress duplicate alerts. This might involve checking existing alerts to see if a new alert matches criteria of an existing one.
6. **Set Actions**:
   * Define the actions to take when the alert is triggered. Actions can include notifying teams, logging the alert, or triggering workflows.
7. **Testing and Tuning**:
   * After setting up the rules, test them to make sure they work as expected.
   * Monitor the alerts to ensure they are being correctly grouped and duplicates are being suppressed.

**Example Configuration**

Here’s an example of a simple alert rule configuration:

1. **Name**: High CPU Usage Alert
2. **Condition**:
   * Source: Monitoring Tool
   * Event Type: CPU Usage
   * Threshold: > 90%
   * Time Frame: Within 5 minutes
3. **Actions**:
   * Notify: System Admin Group
   * Log: Event Log
4. **Correlation Rule**:
   * Check for existing High CPU Usage Alerts within the last 10 minutes.
   * If found, update the existing alert with the new details instead of creating a new one.

By following these steps, you can set up alert rules in ServiceNow that help reduce duplicates and streamline your alert management process.